

Docket No. AUS920030501US1

**CLAIMS:**

What is claimed is:

- 5    1.    A method for testing the compatibility of software versions, the method comprising the computer implemented steps of:
- responsive to an installation of a new software module in a data processing system, performing an
- 10    inventory on an existing set of software modules resident in the data processing system;
- referring to a knowledge base of versions of respective software modules to obtain compatibility information for the new software module with the existing
- 15    set of software modules; and
- providing the compatibility information from the knowledge base, wherein the compatibility information is used to determine whether to install the new software module.
- 20
2.    The method of claim 1, further comprising:
- responsive to a first selected user input, testing the new software module in a test data processing system in combination with the existing set of software modules;
- 25    and responsive to a second selected user input, installing the new software module in the data processing system.

Docket No. AUS920030501US1

3. A method for testing the compatibility of software versions, the method comprising the computer implemented steps of:

responsive to an installation of a new software  
5 module in a data processing system, performing an inventory on an existing set of software modules resident in the data processing system;

referring to a knowledge base of versions of  
respective software modules to determine whether the new  
10 software module is known to be compatible with the existing set of software modules; and

responsive to a negative determination, testing the new software module in a test data processing system in combination with the existing set of software modules.

15

4. The method of claim 3 further comprising:

responsive to a determination that the new software module is compatible with the existing software modules, adding a new combination to the knowledge base; and

20 installing the new software module in the data processing system.

5. The method of claim 3 further comprising:

responsive to a determination that the new software  
25 module is not compatible with the existing modules, searching the knowledge base to find a closest match, wherein at least one of the existing modules is removed or replaced with a different version;

Docket No. AUS920030501US1

prompting for the user as to availability of the  
closest match combination; and

responsive to a user input, installing the new  
5 software module and changing the existing modules as  
needed to obtain a compatible combination.

6. A method in a data processing system for monitoring  
software combinations, the method comprising:

10 identifying a software module;  
determining whether information is present for the  
software module;  
if information is absent for the software module,  
installing the software module to form an installed  
15 software module; and  
testing the installed software module.

7. The method of claim 6, wherein the installing step  
comprises:

20 identifying an environment of a client in which the  
software module is to be installed;  
recreating the environment on a test data processing  
system; and  
installing the software module on the test data  
25 processing system to form the installed software module.

Docket No. AUS920030501US1

8. A data processing system for testing compatibility of software versions, the data processing system comprising:

performing means for performing an inventory on an  
5 existing set of software modules resident in the data processing system responsive to an installation of a new software module in a data processing system;

referring means for referring to a knowledge base of versions of respective software modules to determine  
10 whether the new software module is known to be compatible with the existing set of software modules; and

testing means for testing the new software module in a test data processing system in combination with the existing set of software modules responsive to a negative  
15 determination.

9. The data processing system of claim 8 further comprising the computer implemented steps of:

adding means for adding a new combination to the  
20 knowledge base responsive to a determination that the new software module is compatible with the existing software modules; and

installing means for installing the new software module in the data processing system.

Docket No. AUS920030501US1

10. The data processing system of claim 8 further comprising:

searching means for searching the knowledge base to find a closest match, wherein at least one of the  
5 existing software modules is removed or replaced with a different version responsive to a determination that the new software module is not compatible with the existing modules;

prompting means for prompting the user as to  
10 availability of the closest match combination; and

installing means for installing the new software module and changing the existing modules as needed to obtain a compatible combination responsive to a user input.

15

11. The data processing system for monitoring software combinations, the data processing system comprising:

identifying means for identifying a software module;

determining means for determining whether

20 information is present for the software module;

installing means for installing the software module to form an installed software module if information is absent for the software module; and

testing means for testing the installed software  
25 module.

Docket No. AUS920030501US1

12. The data processing system of claim 11, wherein the installing means comprises:

first means for identifying an environment of a client in which the software module is to be installed;

5 second means for recreating the environment on a test data processing system; and

third means for installing the software module on the test data processing system to form the installed software module.

10

13. A computer program product in a computer readable medium for testing compatibility of software versions, the computer program product comprising the computer implemented steps of:

15 first instructions for performing an inventory on an existing set of software modules resident in the data processing system responsive to an installation of a new software module in a data processing system;

second instructions for referring to a knowledge  
20 base of versions of respective software modules to determine whether the new software module is known to be compatible with the existing set of software modules; and

third instructions for testing the new software module in a test data processing system in combination  
25 with the existing set of software modules responsive to a negative determination.

Docket No. AUS920030501US1

14. The computer program product of claim 13 further comprising:

5       third instructions for adding a new combination to the knowledge base responsive to a determination that the new software module is compatible with the existing modules; and

      fourth instructions for installing the new software module in the data processing system.

10

15. The computer program product of claim 13 further comprising:

15       fifth instructions for searching the knowledge base to find a closest match, wherein at least one of the existing modules is removed or replaced with a different version responsive to a determination that the new software modules is not compatible with the existing modules;

20       sixth instructions for prompting for the user as to availability of the closest match combination; and

      seventh instructions for installing the new software module and changing the existing modules as needed to obtain a compatible combination responsive to a user input.

25

16. A computer program product in a computer readable medium in a data processing system for monitoring software combinations, the computer program product comprising:

Docket No. AUS920030501US1

first instructions for identifying a software module;

second instructions for determining whether information is present for the software module;

5 third instructions for installing the software module to form an installed software module if information is absent for the software module; and

fourth instructions for testing the install software module.

10

17. The computer program product of claim 16, wherein the computer program product comprises:

fifth instructions for identifying an environment of a client in which the software module is to be installed;

15 sixth instructions for recreating the environment on a test data processing system; and

seventh instructions for installing the software module on the test data processing system to form the installed software module.

20

18. A data processing system for managing keepalive transmissions, the data processing system comprising:

a bus system;

a communications unit connected to the bus system;

25 a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to perform an inventory on an existing set of software modules resident in the data processing

30



Docket No. AUS920030501US1

- system responsive to an installation of a new software module in a data processing system; refer to a knowledge base of versions of respective software modules to determine whether the new software module is known to be compatible with the existing set of software modules; and test the new software module in a test data processing system in combination with the existing set of software modules responsive to a negative determination.
- 10 19. A data processing system for managing keepalive transmissions, the data processing system comprising:
- a bus system;
  - a communications unit connected to the bus system;
  - a memory connected to the bus system, wherein the
- 15 memory includes a set of instructions; and
- a processing unit connected to the bus system,
- wherein the processing unit executes the set of instructions to identify a software module; determine whether information is present for the software module;
- 20 install the software module to form an installed software module if information is absent for the software module; and test the installed software module.